**Asg 4 – Debugging / Sample Debugging Report**

**Version Control detail:**

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Username | Password | Link |
|  |  |  |  |

**Task 1: Replication Test Case**

**UAT Test Script**

|  |  |
| --- | --- |
| **Script #1: Bug 1** | |
| **Script Name** | When customer checking out, all service charges are reported as $0.00. |
| **Use Case Tested:** | Service charges should be recorded and added correctly to the system database |
| **Test Description:** | When customers check out and after using some services during their stay, the services are reported, but the cost of each is recorded as $0.00 |
| **Pre-conditions** | Booking for room101 (hotel and customer data must exist)  customer must be checked in  A service charge for $15 |
| **Post-conditions** | Room service must be recorded |

**Bug 1 before fixing the bug:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TEST STEPS** | **EXPECTED TEST RESULTS** | **Test status** |
| ***1*** | Select Checkout from the main menu | Prompt for room number | PASS |
| 2 | Enter room number: 101 | Charge:  Room Service: $15  Total: $15 | **Fail** |
|  |  |  |  |
|  |  |  |  |

|  |  |
| --- | --- |
| **Script #2: Bug 2** | |
| **Script Name** | Player loses double their bet when they lose. |
| **Use Case Tested:** | Players should only lose the exact amount that they bet when they lose. |
| **Test Description:** | When a player doesn’t get a match, their balance goes down by twice the amount they bet |
| **Pre-conditions** | ……………. |
| **Post-conditions** | …………….. |

**Bug 2 before fixing the bug:**

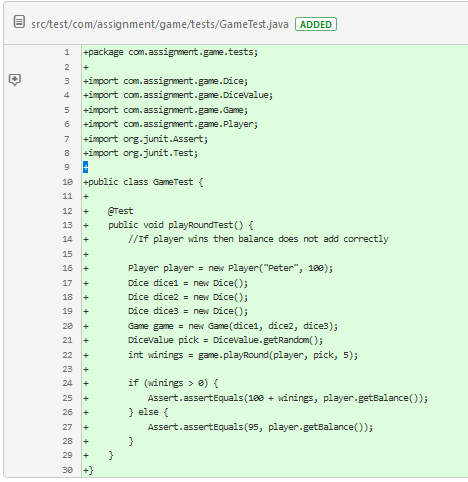
|  |  |  |  |
| --- | --- | --- | --- |
|  | **TEST STEPS** | **EXPECTED TEST RESULTS** | **Test status** |
|  |  |  |  |
|  |  |  |  |

**Task 2: Simplification**

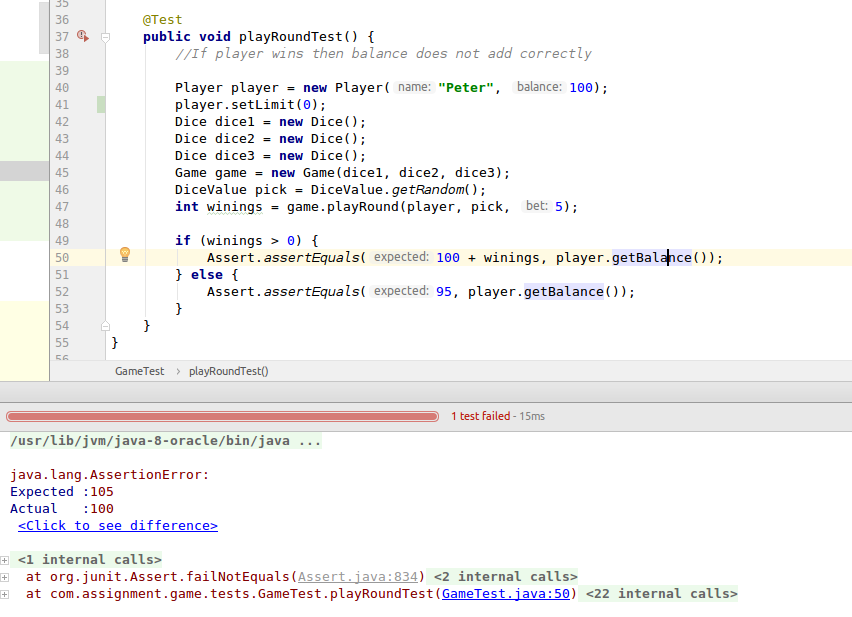
[Note: In this section, unit tests need to be done for the two bugs. Unit test need to be committed to the bitbucket and the output of the test need to be documented here.]

**BUG 1:** **All service charges are reported as $0.00 when checking out**

**Unit test screenshot:**



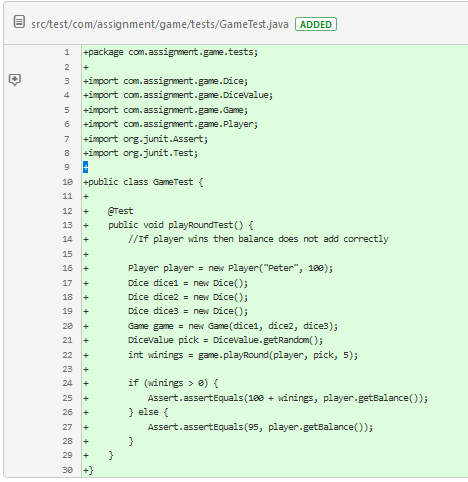
**Unit test result:**



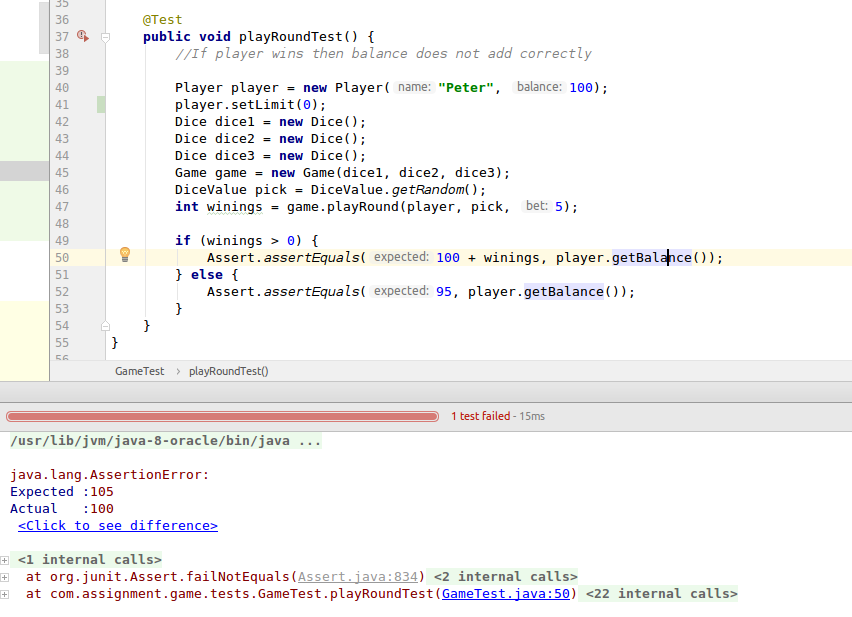
This bug exists because of the error in the ………..

**BUG 2: Player loses double their bet when they lose.**

**Unit test screenshot:**



**Unit test result:**



This bug exists because of the error in the ………..

**Task 3: Tracing**

1. For each of the reported bugs, trace back through the program’s dependences from the point of failure to find the origin of the bug. Use a sequence of hypotheses and tests to support this search and document this sequence in a debugging log.
2. For each of the reported bugs, demonstrate that before the identified origin, variable values are ‘sane’ and that after the origin, at least one value is ‘infected’.
3. Record screenshots demonstrating the origin of the bug.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bug no | Description | Origin/ variables infected | Tracing Dependencies | Hypothesis | Screenshot # |
| 1 | All service charges are reported as $0.00 when checking out . | Error:  **public …**() {  …..  } |  | H0: var Y is reported incorrectly  T0: set a breakpoint at line XX, visually inspect varY value before output.  Result0: H0 false: varY is reported correctly. | 1 |
|  |  | Error:  **public …**() {  …..  }  After fix the error:  **public …..** () {  **….**  ……  } |  | H1: var Z is reported incorrectly  T1: set a breakpoint at line XZ, visually inspect var Z value before output.  Result1: H1 TRUE: var Z is reported incorrectly. | 2 |
| 2 | Player loses double their bet when they lose |  |  |  |  |

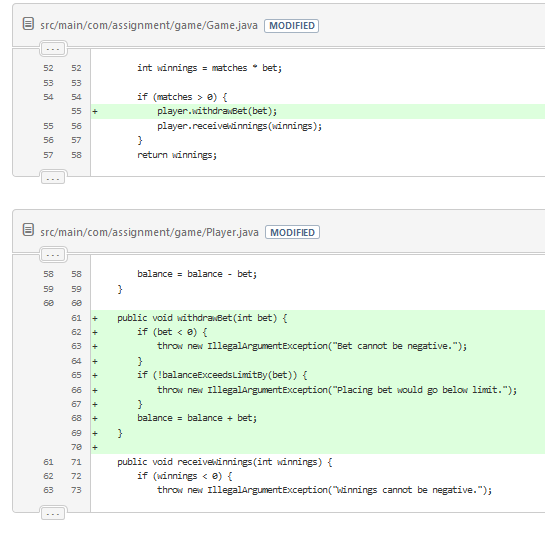
**Task 4: Resolution**

1. For each of the reported bugs, correct the bug.
2. Commit the corrected code to version control.
3. Record output listings from your automated test demonstrating that the buggy behaviour has been eliminated.
4. Carry out the user test case with the specified data and show that the buggy behaviour has been eliminated.

Bug 1:

To fix this problem we have to change/add/remove ……….

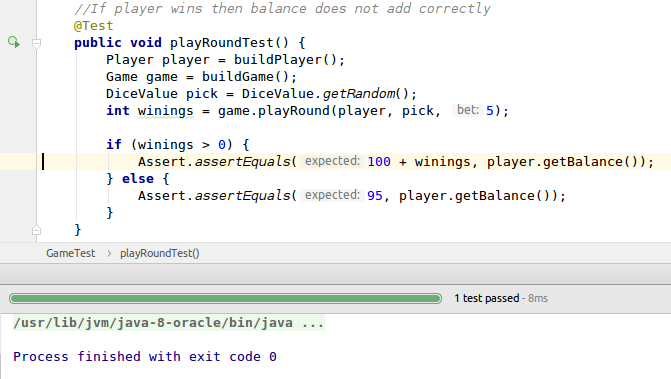
**The code after fixing the bug:**



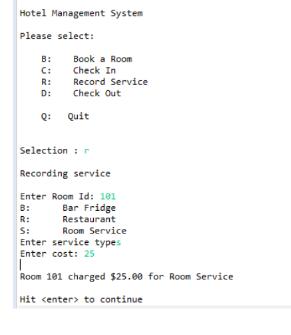
**UAT after fixing bug 1:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TEST STEPS** | **EXPECTED TEST RESULTS** | **Test status** |
| ***1*** | Select Checkout from the main menu | Prompt for room number | PASS |
| 2 | Enter room number: 101 | Charge:  Room Service: $15  Total: $15 | **PASS** |
|  |  |  |  |
|  |  |  |  |

**Unit test result UAT after fixing bug 1: Passed**



**Outputs of program after bug 1 fixed:**



**UAT after fixing bug 2:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TEST STEPS** | **EXPECTED TEST RESULTS** | **Test status** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Unit test result UAT after fixing bug 2: Passed**

**Outputs of program after bug 2 fixed:**